

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application.

**Listing of Claims:**

1. (Currently Amended) A method of updating tunables used in a kernel which remain persistent across computer reboots, comprising:  
updating a system file including tunables each having a tunable setting with a new tunable value in response to a single administrator request;  
simultaneously updating a persistent storage mechanism including tunables each having a tunable setting with the new tunable value in response to the single administrator request;  
changing a tunable value in the kernel with the new tunable value without needing any centralized interface changes and continuing to run the computer with the updated tunable value.
2. (Original) The method of claim 1, wherein said updating steps are performed using a system administrator's management application.
3. (Original) The method of claim 1, wherein updating steps are performed using a UNIX command line.
4. (Original) The method of claim 1, wherein the computer includes a UNIX operating system.
5. (Original) The method of claim 1, wherein the system file includes a file for a core kernel and another for each separately loadable kernel module.

6. (Original) The method of claim 1, further comprising compiling the kernel and incorporating the new tunable value into the kernel.

7. (Original) The method of claim 1, wherein the persistent storage mechanism is a Kernel Registry Service.

8. (Original) The method of claim 1, comprising retrieving a current value of a tunable setting.

9. (Original) The method of claim 1, comprising retrieving detailed information about one or more tunable settings.

10. (Original) The method of claim 1, comprising registering handler functions for a tunable handler for a particular tunable.

11. (Original) The method of claim 1, comprising interfacing with the kernel using a handler function.

12. (Currently Amended) A method of rebuilding a kernel, comprising:  
retrieving tunable settings stored in a system file;  
rebuilding the kernel using the retrieved tunable settings; and  
updating a persistent storage mechanism without needing any centralized interface changes using the retrieved tunable settings.

13. (Original) The method of claim 12, further comprising:

erasing data stored in a persistent storage mechanism;

replacing the erased data with the retrieved tunable settings.

14. (Currently Amended) A dynamic kernel tunable framework for changing tunables in a kernel without rebooting, comprising:

a graphical user interface for displaying and changing graphical values and settings of dynamic tunables;

a system call interface for interfacing the user interface with a system file, a persistent storage mechanism and the kernel;

a handler function interface interfaced to the system call interface and the kernel including information about each dynamic tunable without needing any centralized interface changes.

15. (Original) The framework of claim 14, in which tunable changes are made immediately without rebooting and are also kept persistent across reboots, both actions taken as the result of a single administrator request.

16. (Currently Amended) A computer architecture, comprising:

updating means for updating a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request;

updating means for simultaneously updating a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request; and

changing means for changing a tunable value in the kernel with the new tunable value without needing any centralized interface changes and continuing to run the computer with the updated tunable value.

17. (Currently Amended) A computer architecture, comprising:

retrieving means for retrieving tunable settings stored in a system file;

rebuilding means for rebuilding the kernel using the retrieved tunable settings; and

updating means for updating a persistent storage mechanism without needing any centralized interface changes using the retrieved tunable settings.

18. (Currently Amended) An article, comprising:

at least one sequence of machine executable instructions;

a medium bearing the executable instructions in machine readable form, wherein execution of the instructions by one or more processors causes the one or more processors to:

retrieve a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request;

simultaneously retrieve a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request; and

change a tunable value in the kernel with the new tunable value without needing any centralized interface changes and continuing to run the computer with the retrieved tunable value.

19. (Currently Amended) An article, comprising:

at least one sequence of machine executable instructions;

a medium bearing the executable instructions in machine readable form, wherein execution of the instructions by one or more processors causes the one or more processors to:

retrieve tunable settings stored in a system file;

rebuild the kernel using the retrieved tunable settings; and

update a persistent storage mechanism using the retrieved tunable settings without

needing any centralized interface changes.

20. (Currently Amended) A computer system, comprising:

a processor; and

a memory coupled to said processor, the memory having stored therein sequences of instructions, which, when executed by said processor, causes said processor to perform the steps of:

update a system file including tunables each having tunable setting with a new tunable value in response to a single administrator request;

simultaneously update a persistent storage mechanism including tunables each having tunable setting with the new tunable value in response to the single administrator request; and

change a tunable value in the kernel with the new tunable value without needing any centralized interface changes and continuing to run the computer with the updated tunable value.

21. (Currently Amended) A computer system, comprising:

a processor; and

a memory coupled to said processor, the memory having stored therein sequences of instructions, which, when executed by said processor, causes said processor to perform the steps of:

update tunable settings stored in a system file;

rebuild the kernel using the retrieved tunable settings without needing any centralized interface changes; and

update a persistent storage mechanism using the retrieved tunable settings.

22. (New) The method of claim 1, wherein the tunables can be made independently of each other.

23. (New) The method of claim 1, wherein new tunables can be created without needing any centralized interface changes.

24. (New) The method of claim 1, wherein a data structure is maintained that includes information about every tunable parameter.

25. (New) The method of claim 24, comprising compiling the data structure into the kernel to initialize the data structure in a kernel registry.

26. (New) The method of claim 1, wherein each tunable has an associated handler function.

27. (New) The method of claim 1, comprising updating a tunable using SAM.

28. (New) The method of claim 1, comprising updating a tunable using kmtune.

29. (New) The method of claim 1, comprising updating a tunable using a kernel system call (API).

30. (New) The method of claim 1, comprising using one of a SAM, kmtune and a kernel system call to update a tunable.